

# failing to predict en passant :(

Leon Lee and Lila Marshman

April 10, 2023

## 1 Overview

Online chess sites receive a huge amount of game data from large volumes of users accessing and playing on their sites. Many have large player bases, representing a huge variety in player Elo scores (a number corresponding to skill level). We used data from games played on the online chess website Chess.com to analyse the relevancy of players' Elo in a chess game. We used a logistic regression model to investigate the effect of the difference in Elo on winning a game. Following from this, by using an exponential function "Temptation", we were able to visualise a player's likelihood of playing a specific chess tactic called *en passant* across three different categories of Elo. We then attempted to predict a player's Elo using the context of said *en passant* move. By analysing computer generated evaluations of the state of the game preceding the move, and by using Temptation as a hyperparameter for a predictive model, we attempted to utilise PCA and a K-Nearest Neighbour algorithm to classify users into three classes of Elo. To evaluate the model we calculated various metrics (accuracy, precision, recall, F1 score). [Lila Talk about evaluation and results]

All decimal values in this report are provided to 4 decimal places.

## 2 Introduction

**Context and motivation** Online chess sites such as Chess.com allow users to play with friends or strangers, offering a wide variety of chess variants and time controls to play with. Online chess' rise in popularity follows the increase in free time during the COVID-19 pandemic lockdowns, the popularity of Netflix's show 'The Queen's Gambit', and world-ranking players streaming the game on Twitch [3] With a sudden increase in online players comes an increase in publicly available game data - this provides a perfect opportunity for an investigation into player's skill (measured by Elo points). In this study, we explore the impact of a player's skill level on a game's direction and patterns in the play styles of players at different Elos.

The particular play style we explore in this study is the context surrounding an *en passant* chess move. Typically, chess tactics are something you aim for, therefore there is the natural assumption that a higher-ranked player would be able to set up certain tactics more consistently than a lower Elo one. One exception to this is the move *En passant*. It is an incredibly situational move, and it heavily relies on a player's opponent to move a certain way for a player to be able to play it in the first place. In the chess scene, particularly in the online chess community, *en passant* has gained a cult-like following[5]. It's become a popular running joke amongst players to always capture via *en passant* when given the chance, even if this puts them in a worse

position than having not chosen that move. Losing an online game results in your Elo rating decreasing, thus most would expect highly rated players to not risk a bad game position, hence not capture *en passant* unless it's beneficial. In this study we explore whether we can use the context surrounding an *en passant*-allowed board state to predict whether a player falls into the low, medium, or high Elo category.

Insight into these areas, particularly if the second investigation proves Elo is predictable from move contexts, may be useful in determining how players in the past compare to today's players. There is much speculation on how historical chess champion Bobby Fischer would compare to current world champion Magnus Carlsen [9], thus if we are able to predict a modern-day Elo for Fischer by inputting his play style information into a model trained on modern games, we may find evidence suggesting how he'd compare.

**Previous work** Brief description of any previous work in this area (e.g., in the media, or scientific literature or blogs).

LILA REMOVE THIS (just here so I can remember tomorrow):

A book by Holding [7] suggests that Elo ideally follows a normal distribution with a mean of 1,500 and a standard deviation of 200 - however this fluctuates as the population of rated chess players over time. Holding

[7] proposes that if Elo follows this distribution exactly, a player rated at the mean has a 0.25 chance of winning against players rated at 1,700, and a 0.1 chance of winning against players rated 1,875, though it is unclear how these probabilities were calculated. This implies there is some known association between comparative Elo and winning, though no further explanation on this is given.

An article published on the website 'Towards Data Science' [11] found a relationship between Elo disparity and winning within a range of  $\pm 50$  Elo points difference, but outside of this range an increase in Elo had little effect on increasing or decreasing the probability of a win. [11] created a logistic regression model to predict winning probability for data points within the  $\pm 50$  Elo points difference. They demonstrated there was a correlation, however, they do not explore logistic regression coefficients, interpret their visualisations, or evaluate their model any further than stating their AUC metric is "not bad". This provides the opportunity for us to explore data outside the  $\pm 50$  Elo difference range, as well as running hypothesis tests and interpreting regression coefficients to further justify [11]'s findings. There is no literature directly studying the relationship between a player's response to *en passant* moves and their Elo.

**Objectives** Our goals in this study are to investigate whether there is a correlation between a player's Elo, and the moves they play in a game. In our case we are focusing on the move *en passant*, and a player's willingness to play the move if an opportunity presents itself. We will first take a bird's eye view of the overall effects of Elo, and investigate whether a significant gap in Elo affects the chances of winning. Then, we will analyse games containing the move *en passant*, and through the context of the board state try and determine a player's Elo, and indirectly, their chance of winning.

### 3 Data

**Data provenance** We obtained our data from Kaggle.com [1], where they provided a dataset of over 60,000 games of chess taken from Chess.com. The User Agreement on Chess.com states that you are not allowed to data mine [4], but in this case the dataset was extracted using the Chess.com API so it complies with their regulations.

**Data description** The dataset records 66,879 games of chess that took place on Chess.com with varying game modes, time classes, and levels of players. Information about each player is provided, i.e. the usernames, Chess.com profiles, and Elo rating during the match. Information about the game is also provided, i.e. the result, information about the time rules, whether it was rated, and the final chess board in a notation called "FEN". The final column is called the PGN (Portable Game Notation)

- a column in a standard format to be easily read by other chess analysers. Within the PGN, there includes a list of the moves that took place during the game. Using this we can simulate and replay exactly how the game was played, and provide further analysis.

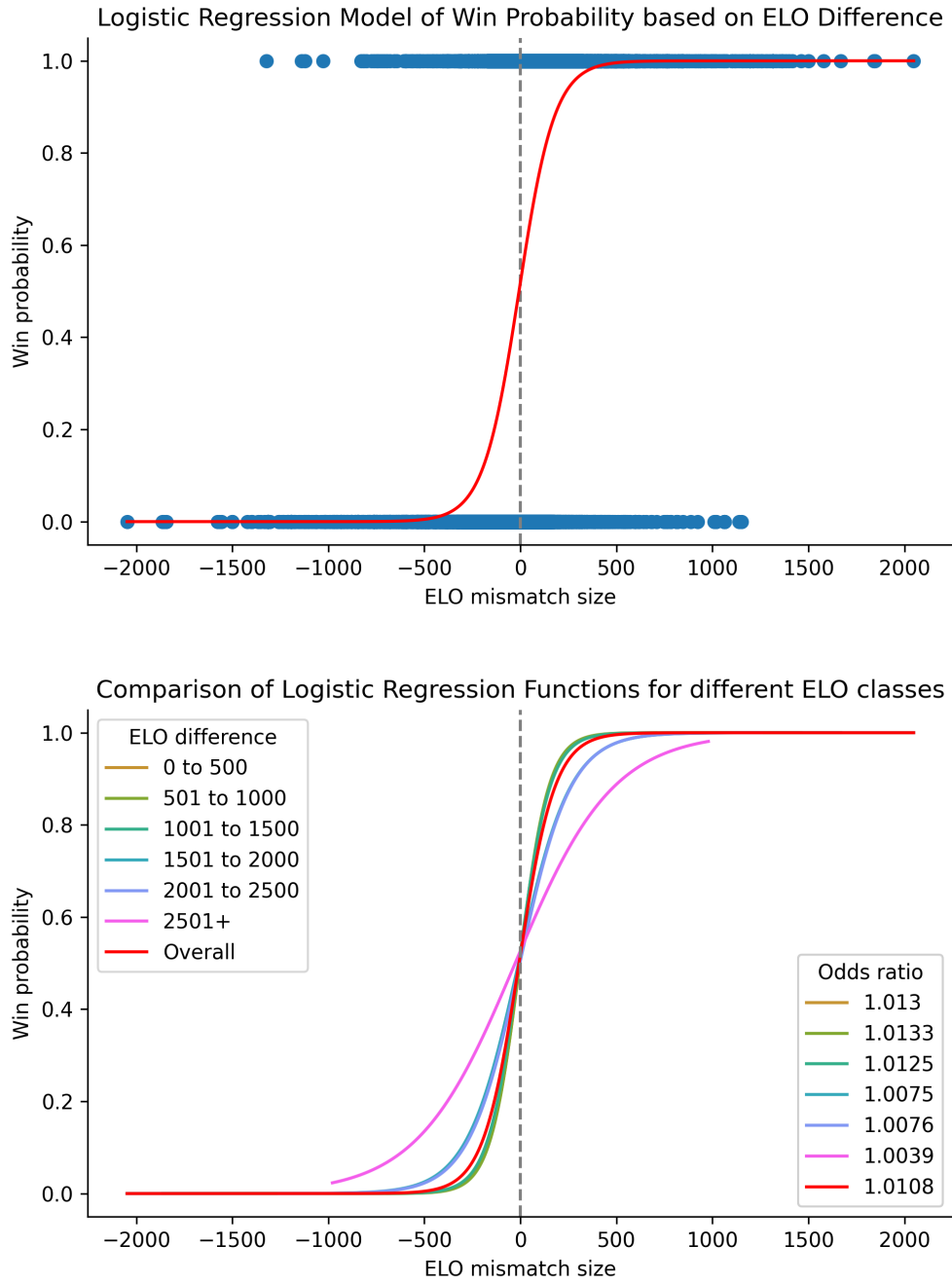
**Data processing** For both investigation questions, we removed alternate game modes that were not standard chess (for example, 'Chess960', and all game records if they contained any NaN values. We removed games where the game terminated early by using two filters: determining no major pieces had been moved (by reading the top row of the 'FEN', and removing games with less than 10 total moves (5 per player). We believed any games in these two categories would not be useful to the data analysis. Additionally, we removed games from the 'daily' time class. This was due to these records' 'PGN' column being structured in a different format, meaning daily games would have needed to be analysed separately from other modes. Additionally, unlike the rest of the remaining games, 'daily' games had no time pressure on players - thus removing these removes the chance that time pressure may have affected the results as a confounding variable. Due to this, and the fact there was only a small amount of 'daily' games ( $\sim 9.2\%$  of total games), we removed these from the dataset.

For the first investigation, we required a binary outcome (winning or losing) in order to use logistic regression, thus we additionally excluded all games which ended in a draw or other indeterminate end-states.

For our second investigation, we utilised various external libraries to assist in defining k-NN hyperparameters. Using the *python-chess* library [12], we parsed the "PGN" field to filter out games that didn't include an *en passant* opportunity ( $n = 5074$ ), and highlight games where an *en passant* move actually occurred ( $n = 1563$ ). We then used *Stockfish*[13], an open-source chess game engine, to evaluate how much of an advantage from an initial board state a player would gain from the following moves: a Stockfish calculated best move, the move the player decided to make, and finally the relevant *en passant* capture. These scores were added to the dataframe. All non-numerical data were converted to numerical data by assigning numerical categories. This was important for standardising the data, performing a PCA and creating the k-NN model.

## 4 Exploration and analysis

**Data Analysis: Question 1 - Investigating the relationship between players' Elo difference and winning** LILA MAKE SURE NUMBERS GIVEN IN THIS TEXT MATCH WITH THE ONES GIVEN ON THE GRAPH, EG BETA1



**Figure 1:** Subplot 1: Each datapoint represents how many Elo points a player is than their opponent during a game, and whether they won (1) or lost (0). The standard logistic function is plotted in red. 'Elo mismatch size' is a measure of the number of Elo points a player's opponent is above them.

Subplot 2: Each line represents the logistic function plotted for a particular Elo difference class. Information on the odds ratios for each is provided, calculated from  $e^{\beta_1}$  for each class' regression coefficient  $\beta_1$ . The red line shows the overall standard logistic function from subplot 1, to aid slope comparisons for each class.

We used logistic regression to investigate the association between the difference in players' Elo and winning. We believed this to be the most appropriate technique to use since logistic regression typically works well for data with a continuous predictor (Elo difference) and a binary response variable (winning or losing). The differences in Elos for each game were calculated from the perspective of a white-playing player. For example, if in a game, white had an Elo of 1000 and black had an Elo of 900, the difference in Elo for this game would be recorded as  $-100$ .

After applying logistic regression to the sample data we visualised the results (Figure 1) and found the regression coefficients  $\beta_0$  and  $\beta_1$  to be 0.0768 and 0.0103 respectively.  $\beta_0$  describes the log odds for opponents of the same Elo. Since it is close to 0 (0.0768 logits), this tells us winning or losing are almost equally likely for players with 0 difference in Elo. This is shown in Figure 1, Subplot 1 where we see the logistic function has an almost 0.5 win probability where a player's opponent is 0 Elo points higher than them. We used  $\beta_0$  to calculate this probability exactly as

$$\frac{1}{1 + e^{\beta_0}} = 0.48081.$$

Furthermore, the odds ratio  $e^{\beta_1} = 1.0103$  shows that for every 1 point higher a player is in Elo, they are 1.0103 times more likely to win against their opponent. Figure 1, Subplot 1's logistic function line also shows that at around  $\pm 400$  Elo points difference, the outcome is predicted as almost certainly a win for the player with a higher Elo. Figure 1, Subplot 1 also shows that in the sample used, white-playing players against opponents rated 1,500 Elo points lower than them always won. Similarly, white-playing players against opponents rated 1,200 Elo points above them always lost.

Figure 1 Subplot 2 shows the different logistic functions produced for classes of differing Elo, alongside the main model's overall logistic function. The trend for the regression coefficients  $\beta_1$  typically show that the larger the Elo difference, the smaller the odds ratio  $\beta_1$  is. Thus for larger differences in Elo, the rate of increase in likelihood of a player beating their opponent decreases, despite the actual likelihood of beating their opponent increasing.

The logistic regression model was estimated using maximum likelihood, so it seemed most appropriate to use a Wald test to test for a relationship between Elo difference and the probability of a win [6]. We used the null hypothesis  $H_0 : \beta_1 = 0$  which states there is no statistically significant relationship, and alternative hypothesis  $H_a : \beta_1 \neq 0$  which suggests there is a statistically significant relationship. Following the test procedure outlined by Forthofer, Lee and Hernandez [6], and a method to calculate the Wald statistic inspired by StackExchange user j\_sack [8], we found a Wald statistic of 75.3946,

which gave a p-value of  $p < 0.01$ . Thus we may reject the null hypothesis at the 1% level, concluding there is sufficient evidence to reject the notion that there is not a statistically significant relationship between Elo difference and probability of winning.

## Data Analysis: Question 2 - Using context of an en-passant move to attempt to predict Elo of a player

To represent the different categories of a player's Elo, we ordered Elo by size and divided our dataset into roughly three equal-sized Elo classes. Since player Elos were normally distributed, we used the sample mean ( $\bar{X} = 1,240$ ) and standard deviation ( $\sigma = 400$ ) to derive these Elo classes such that the amount of data classed in each low, medium and high Elo group was in an almost 30 : 40 : 30 ratio. We believed this ratio most effectively balanced the effects of a large standard deviation whilst ensuring there wasn't a huge difference in the amount of data categorised in each class. Thus the resulting class boundaries became: 0 – 1000 Elo, 1001 – 1400 Elo, and greater than 1401 Elo. Filtering out games where en-passant capture didn't happen, we plotted the distribution of each class on different levels of temptation (Fig. 2). We defined the function "temptation" to estimate how viable a player would find an en-passant move, modelled as such:

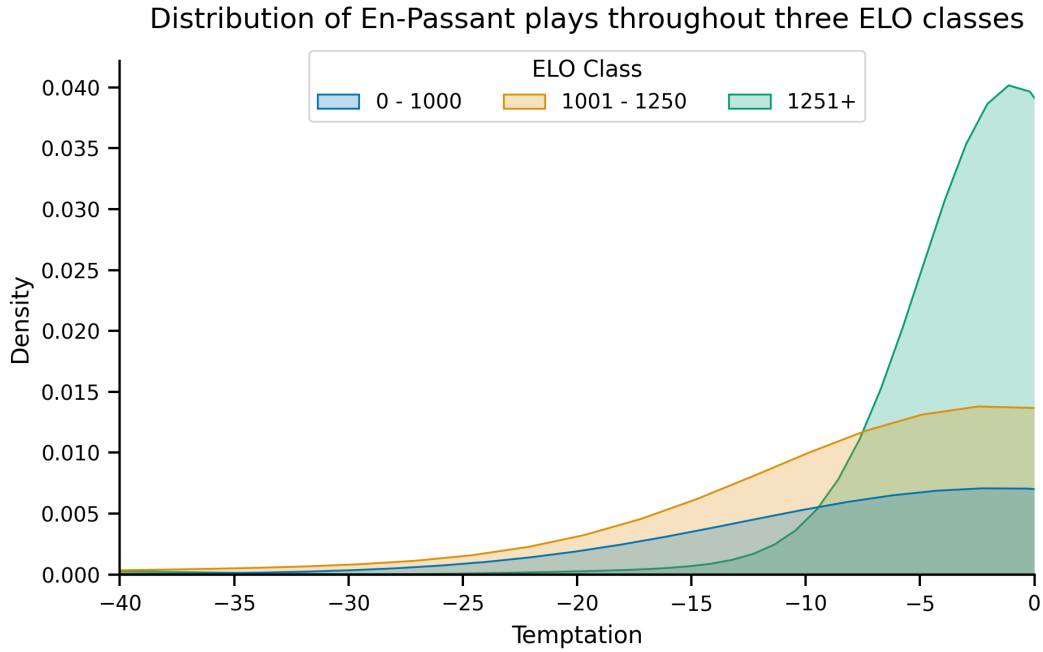
$$T(\text{move}) = 1 - e^{(E_{\text{best}} - E_{\text{move}})/n}$$

This is a very basic model but it ultimately works off the basis that for a player making a move, a very badly evaluated move would be significantly less likely to be chosen than a move close in evaluation to the best move. The number  $n$  is a factor to stop the exponential function growing too fast, and no significant value of  $n$  proved to be significantly better than others so we chose  $n = 220$ .

After plotting this graph, we tried to emulate these results by creating a new field for the probability that someone in a certain Elo would play an en-passant move. There wasn't enough samples in the dataset to get any meaningful information from comparing single values, so we rounded each player's Elo to the nearest 10 and calculated the probability using the following formula:

$$\mathbb{P}(\text{Elo class}) = \frac{\text{Games with successful e.p. captures}}{\text{Total games with e.p. opportunity}}$$

At this point, there are 8 parameters involved in the data, 4 of those are categorical, i.e. the player's colour, if en-passant actually happened, if the game was rated, and the time class. The other variables being the time taken, a Stockfish evaluation of the player's actual move in the dataset, the probability of someone in their Elo class playing en-passant, and finally the temptation. We then split our dataset in a 60 : 40 ratio for the training and test set respectively (Fig. 3). We used this ratio because any wider a difference and the model would overfit/underfit the data [Lila work out which one it is] meaning the



**Figure 2:** Every value of temptation on the chart is negative since we assume a move cannot be better than the Stockfish evaluated best move. Values with a temptation of 0 were excluded as the number of games where en-passant did occur if it was the best move was overwhelmingly high across every class.

confusion matrix wouldn't [Lila finish this]. Using the training set, we performed a PCA analysis to reduce the dimensionality of the dataset, and then utilised a k-nearest neighbours algorithm to classify the test set.

From Fig. 2, we can see that higher Elo players have a very concentrated area of temptation where they will perform a capture, i.e. when the temptation is close to 0. The drop-off after being relatively sharp, and turning to a near-zero chance for any move below  $T(\text{move}) = -20$ . On the other hand, both the medium and lower Elo classes have a much spread-out and shallower slope, which indicates that in lower Elos, the novelty of playing en-passant does outweigh the potential repercussions of playing a bad move. However, all three Elo classes seem to follow a normal distribution, implying it would be possible to predict how a player at a certain ELO will play given an en-passant opportunity.

On the other hand, from the K-NN Chart (Fig. 3), the left and right sides are somewhat defined, but the middle Elo class has a very erratic classifying area, with it being blended in with the other two classes as well. This is likely due to there being a lot of overlap from all three categories in the section where most medium Elo games lie, making it difficult to accurately classify a data point inside of that region from the parameters that we used. From these two graphs, it is clear that although we can somewhat predict the chances of an en-passant move given the Elo of a player, the inverse is not true and using only context surrounding an en-passant move

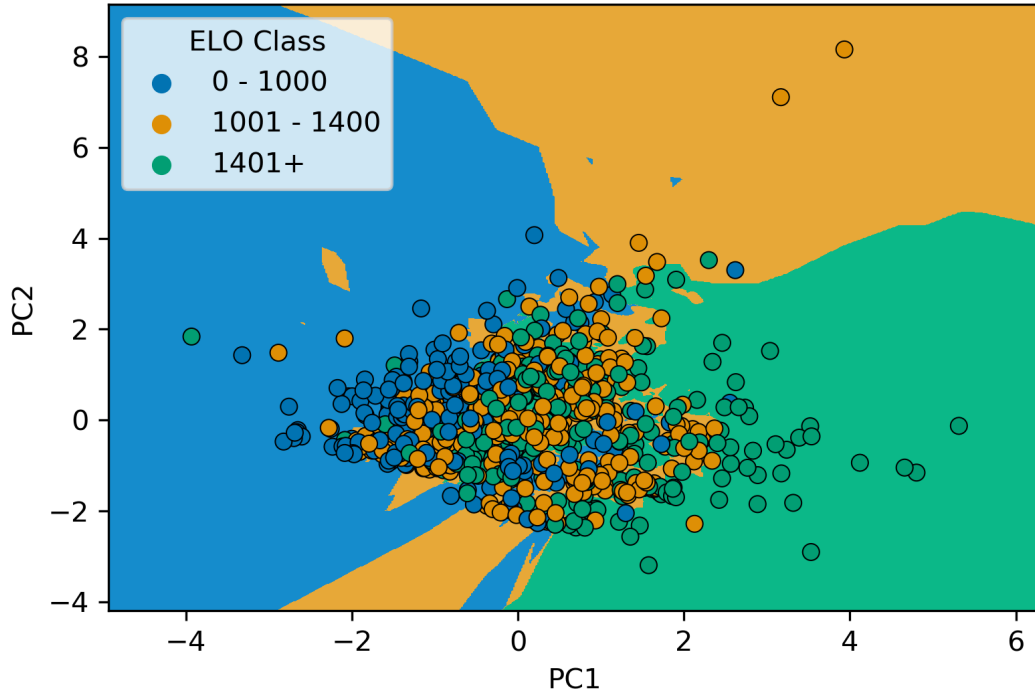
is not nearly enough to predict someone's Elo without a much more accurate model and further parameters of the players involved.

To further back this up, we created a confusion matrix (Table 1) detailing the numbers of correct and incorrect predictions of the k-NN model on the test set for each Elo class.

Values from the confusion matrix were used to calculate metrics to evaluate the model's predictive performance. These metrics were: accuracy (0.3412), precision (0.3991), recall (0.3412), and sample-weighted F1 score (0.3544). The accuracy is below 0.5, thus demonstrating our model has poor predictive capabilities. The precision and recall scores were used in the calculation for the F1 score, which evaluates the model for precision (amount of correctly classified data points) and robustness (whether it misses a significant amount of data) [10]. To account for the test set containing different proportions of each Elo class, we used the sample-weighted calculation for the F1 score. This produced an F1 score of 0.3544 which shows a bad fit of the model to the data, implying Elo cannot be accurately predicted from the model's hyperparameters.

We further produced an ROC [Lila do this and cite figure]

K-Nearest Neighbours of PCA Graph, classifying three Elo classes



**Figure 3:** Each dot in the chart represents a unique player in the dataset, and their Elo is sorted into the corresponding category of Elo class. The algorithm is tuned with between 3 and 7 neighbours and picks the result with the highest accuracy, in this case 5.

**Table 1:** Confusion matrix of the test set, providing information on the number number of data points the model predicted to be in each Elo class, alongside the actual amount in each Elo class. Here Elo classes are coded as: 1 : 0 – 1,000, 2 : 1,001 – 1,400, 3 : 1,401+.

Location	Predict 1	Predict 2	Predict 3
Actual 1	10	95	5.3
Actual 2	3	40	5.3
Actual 3	0	10	6.3

## 5 Discussion and conclusions

**Summary of findings** This study’s findings indicate that the amount of difference in Elo between two players affects the outcome of a chess game, for games played on the online chess website Chess.com.

By focusing on the situational board state whenever an *en passant* chess move was played, we developed a k-NN model to determine whether there was enough information from this context to predict the Elo of the player. However we found insufficient evidence that a

player’s Elo was predictable from the context of this one move.

### Evaluation of own work: strengths and limitations

A caveat of using Stockfish is that since it is a live engine, without utilising significant processing power it will end up with slightly different results each time it runs. Although we have found that in most cases the data is similar, sometimes it generates a large variation in the graphs that will be shown in this study.

Another limitation is that while the "quotient" field measuring the probability of a player playing an *en passant* move cannot be used backwards to predict the Elo of the player, the derivation of the value is taken from a rounding of the player’s Elo. So this study is more of a proof of concept, as it cannot be viable to predict a person’s Elo blind. A more viable option is to instead of grouping by similar Elo class, to group by individual players instead. However, in this dataset there is not a large enough sample to viably do this, as the mean number of games with a player is only 1.7370, with a standard deviation of 4.0367.

**Comparison with any other related work** E.g. “Anscombe has also demonstrated that many patterns of data can have the same correlation coefficient” [2].

Wikipedia can also be cited but it is better if you find the original reference it for a particular claim in the list of references on the Wikipedia page, read it, and cite it.

**Improvements and extensions** The model used to predict the "temptation" of a move is very simplistic and does not consider other parameters other than the evalu-

ations given by Stockfish. In the real world, there would be lots of other factors such as remaining time, or the fact that a human would not evaluate moves in the same way as a computer. However, in Fig. 2, the results that were displayed were what we were expecting in this study. If improved and tuned better, the idea for the model could be used to analyse moves beyond just en-passant.

## References

- [1] ADITYAJHA1504. *60,000+ Chess Game Dataset (Chess.com)*. Last accessed 19th March 2023. 2021. URL: <https://www.kaggle.com/datasets/adityajha1504/chesscom-user-games-60000-games>.
- [2] Francis J Anscombe. "Graphs in statistical analysis". In: *The American Statistician* 27.1 (1973), pp. 17–21.
- [3] Christian Behler. "The 2020 Chess Boom". In: *SUPERJUMP* (2020). Retrieved 19 March 2023. URL: <https://medium.com/super-jump/the-2020-chess-boom-992427704a28>.
- [4] Chess.com. *Chess.com Terms of Service*. Last accessed 3rd April 2023. URL: <https://www.chess.com/legal/user-agreement>.
- [5] Chess.com. *En passant*. Last accessed 9th April 2023. URL: <https://www.chess.com/article/view/en-passant-awareness#memes>.
- [6] Ronald N. Forthofer, Eun Sul Lee, and Mike Hernandez. *Biostatistics*. Second Edition. Elsevier Inc., 2007.
- [7] D.H. Holding. *The Psychology of Chess Skill*. Psychology Revivals. Taylor & Francis, 2021. ISBN: 9781000394788. URL: <https://books.google.co.uk/books?id=-tlGEAAAQBAJ>.
- [8] j\_sack and AllanLRH. *How to compute the standard errors of a logistic regression's coefficients*. Last accessed 2nd April 2023. 2016. URL: <https://stats.stackexchange.com/questions/89484/how-to-compute-the-standard-errors-of-a-logistic-regressions-coefficients>.
- [9] Giovanni Di Luca. *Bobby Fischer Vs Magnus Carlsen: Who Is Really The Best Ever?* Last accessed 9th April 2023. 2021. URL: <https://chesspulse.com/bobby-fischer-vs-magnus-carlsen-who-is-the-best-ever/>.
- [10] Aditya Mishra. "Metrics to Evaluate your Machine Learning Algorithm". In: *Towards Data Science* (2018). Retrieved 6 April 2023. URL: <https://towardsdatascience.com/metrics-to-evaluate-your-machine-learning-algorithm-f10ba6e38234>.
- [11] Eric Onofrey. "How much does Elo Matter?" In: *Towards Data Science* (2019). Retrieved on 19 March 2023. URL: <https://towardsdatascience.com/how-much-does-elo-matter-7e8c3e910cb1>.
- [12] *python-chess 1.999*. Last accessed 9th April 2023. URL: <https://pypi.org/project/python-chess/>.
- [13] *Stockfish - Open Source Chess Engine*. The latest version of Stockfish is V15.1, but we used Stockfish V14 for better compatibility with UoE DICE machines. URL: <https://stockfishchess.org/>.